

---

## 5. Lawn Service by Ronnie

**Program Name:** Lawn.java

**Input File:** lawn.dat

Ronnie has a lawn service. When figuring an estimate, he needs to know what part of the yard he can mow with his big mower and what part he has to mow with a smaller mower. His big mower covers a 36" square but can only move horizontally or vertically. Any spot in the yard that he cannot get his big mower into, he must mow using the smaller mower. To help him with his estimates, he has made a rectangular matrix of each yard.

Each cell of his matrix represents a one foot by one foot square of the yard that is either an area that needs to be mowed or an area that is not to be mowed. He cannot mow areas where the entry into the area is less than 36" and he may be required to mow over some areas twice (or more). You are to write a program that will show the parts of the yard he can mow with the big mower and the parts he must mow using the smaller mower.

### Input

The first line of input will contain a single integer  $n$  that indicates the number of yards he plans to mow. The first line of each yard will contain two integers  $r$   $c$  that denote the number of rows and columns respectively in the lawn. The following  $r$  lines will contain  $c$  characters with no spaces. The characters are:

- A period (.) denoting an area to be mowed.
- An asterisk (\*) denoting an area to mowed around like a tree or flower bed.
- An equal sign (=) denoting the starting cell which will always be in the upper left corner of the matrix.

### Output

You will output the matrix for each yard with all periods replaced by:

- A B if the area can be mowed by the big mower
- An s if the area must be mowed by the small mower.

Print a blank line after each matrix.

### Example Input File

```
2
5 15
=. . . . . . . . . . . . . . .
. . . * . . . . * . . . * .
. . . . . . . . . . . . . . .
. . . . . . . . . . . . . . .
. . . . . * . . . . * . . . .
6 20
=. . . . * . . . * . . . . . . .
. . . . . . . . . . . . . . .
. . . . . . . . . . . . . . .
. . . . . . . . . * . . . . .
. . . * . . . . . . . . . * . .
. . . . . . . . . . . * . . .
```

### Example Output to Screen

```
BBBssssssssssss
BBB*sssss*sss*s
BBBBBBssssssss
BBBBBBssssssss
BBBBBB*sssss*ss

BBBBB*BBB*BBBBBBBBBB
BBBBBBBBBBBBBBBBBBBB
BBBBBBBBBBBBBBBBBBBB
BBBBBBBBBBBBBB*BBBBBB
BBB*BBBBBBBBBBsBBB*ss
BBBsBBBBBBBBBBsBBB*ss
```